

LISTING OF CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application:

1. (Previously Presented) Method for managing compiled filter code for processing data packets wherein compiled filter code is managed in a plurality of pieces, said pieces are pages having a predetermined length and shadow paging is used and comprising the steps of processing packets according to at least one first set of code pages, creating a second set of code pages to represent the set of code pages to be used after a certain point in time, processing packets received after said certain point in time according to said second set of code pages, and processing packets received before said certain point in time according to said at least one first set of code pages.
2. (Original) Method according to claim 1 comprising the steps of incrementally compiling at least one rule for obtaining a piece of code by a rule compiling entity, transmission of said piece of filter code from a rule compiling entity to a packet processing entity, pausing of processing of packets by said packet processing entity, writing of said piece of filter code to memory means, and continuing of processing of packets by said packet processing entity.
3. (Original) Method according to claim 1 comprising the steps of incrementally compiling at least one rule for obtaining a piece of code by a rule compiling entity, signalling from said rule compiling entity to a packet processing entity that a new piece of code is compiled, signalling from said packet processing entity to said rule compiling entity that said packet processing entity is ready for storage of said piece of code, writing said piece of code to a memory means, and signalling from said rule compiling entity to said packet processing entity that said piece of code is written to said

memory means.

4. (Canceled)

5. (Canceled)

6. (Canceled)

7. (Previously Presented) Method according to claim 1 comprising within said step of creating a second set of code pages the steps of assigning members of an existing code page set to be members of said second set of code pages, and removing a code page from said second set of code pages.

8. (Previously Presented) Method according to claim 1 comprising within said step of creating a second set of code pages the steps of creating a new code page, and assigning said new code page to be a member of said second set of code pages.

9. (Previously Presented) Method according to claim 1 comprising the step of removing a code page from the memory element storing the code pages, when the code page is not any more a member of any set of code pages in use.

10. (Previously Presented) Method according to claim 1 wherein each page of code is

associated with a reference number for observing the order of the code pages.

11. (Previously Presented) Method for managing compiled filter code for processing data packets wherein compiled filter code is managed in a plurality of pieces, each page of code is associated with a reference number for observing the order of the code pages, wherein the order of any two code pages is determined by comparing values of $v(x)$ calculated from the reference numbers associated with the code pages, $v(x)$ being calculated substantially by the formula $v(x)=r(x)-r(b)\text{mod}M$ where $r(x)$ is the reference number associated with a code page x being compared, $r(b)$ the reference number of the base code page, and M the size of the set of allowed reference numbers $\{0, 1, 2, \dots, M-1\}$.

12. (Previously Presented) Computer software program product for processing data packets based on compiled filter code comprising computer program code means for managing the compiled filter code in a plurality of pieces, computer program code means for implementing shadow paging of pages of filter code, computer program code means for processing packets according to at least one first set of code pages, computer program code means for creating a second set of code pages to represent the set of code pages to be used after a certain point in time, computer program code means for processing packets received after said certain point in time according to said second set of code pages, and computer program code means for processing packets received before said certain point in time according to said at least one first set of code pages.

13. (Original) Computer software program product according to claim 12 further

comprising computer program code means for incrementally compiling at least one rule and for producing at least one piece of code, and computer program code means for updating a memory means with said at least one piece of code.

14. (Canceled)

15. (Canceled)

16. (Original) Computer software program product according to claim 12 wherein the computer software program product is a software routine library.

17. (Previously Presented) A computer program comprising instructions adapted for carrying out the steps of the method according to any one of claims 1 to 3 or 7 to 11.

18. (Previously Presented) Computer network node for processing of data packets according to compiled filter code comprising means for managing the compiled filter code in a plurality of pieces and comprising means for processing packets according to at least one first set of code pages, means for creating a second set of code pages to represent the set of code pages to be used after a certain point in time, means for processing packets received after said certain point in time according to said second set of code pages, and means for processing packets received before said certain point in time according to said at least one first set of code pages.

19. (Original) Computer network node according to claim 18 further comprising means for

incrementally compiling at least one rule and for producing at least one piece of code, and means for updating a memory means with said at least one piece of code.

20. (Original) Computer network node according to claim 18 further comprising means for implementing shadow paging of pages of filter code.

21. (Canceled)

22. (Original) Computer network node according to claim 18 wherein the node is a virtual private network node.

23. (Original) Computer network node according to claim 18 wherein the node is a router node.

24. (Original) Computer network node according to claim 18 wherein the node is a firewall node.

25. (Original) Computer network node according to claim 18 wherein the node is a workstation.

26. (Previously Presented) System for processing of data packets according to compiled filter code comprising means for managing the compiled filter code in a plurality of pieces comprising means for processing packets according to at least one first set of code pages, means for creating a second set of code pages to represent the set of code pages to be used after a certain point in time,

means for processing packets received after said certain point in time according to said second set of code pages, and means for processing packets received before said certain point in time according to said at least one first set of code pages.

27. (Original) System according to claim 26 comprising means for incrementally compiling a set of rules and for producing at least one piece of code, and means for updating a memory means with said at least one piece of code.

28. (Original) System according to claim 26 comprising means for implementing shadow paging of pages of filter code.

29. (Canceled)

30. (Original) System according to claim 26 further comprising a memory component having a first access port and a second access port, and means for processing data packets, said means for processing data packets being arranged to access said memory component via said first access port, and said means for managing the compiled filter code being arranged to access said memory component via said second access port.